



**Europäisches  
Patentamt**

**European  
Patent Office**

**Office européen  
des brevets**

PCT/IB 03 / 0 4 3 2 9  
Rec'd PCT/PTO 31 MAR 2003

REC'D 10 OCT 2003

WIPO

PCT

**Bescheinigung**

**Certificate**

**Attestation**

Die angehefteten Unterla-  
gen stimmen mit der  
ursprünglich eingereichten  
Fassung der auf dem näch-  
sten Blatt bezeichneten  
europäischen Patentanmel-  
dung überein.

The attached documents  
are exact copies of the  
European patent application  
described on the following  
page, as originally filed.

Les documents fixés à  
cette attestation sont  
conformes à la version  
initialement déposée de  
la demande de brevet  
européen spécifiée à la  
page suivante.

**Patentanmeldung Nr. Patent application No. Demande de brevet n°**

02079126.5

## **PRIORITY DOCUMENT**

SUBMITTED OR TRANSMITTED IN  
COMPLIANCE WITH RULE 17.1(a) OR (b)

Der Präsident des Europäischen Patentamts;  
Im Auftrag

For the President of the European Patent Office

Le Président de l'Office européen des brevets  
p.o.

**R C van Dijk**



Anmeldung Nr:  
Application no.: 02079126.5  
Demande no:

Anmeldetag:  
Date of filing: 04.10.02  
Date de dépôt:

Anmelder/Applicant(s)/Demandeur(s):

Koninklijke Philips Electronics N.V.  
Groenewoudseweg 1  
5621 BA Eindhoven  
PAYS-BAS

Bezeichnung der Erfindung/Title of the invention/Titre de l'invention:  
(Falls die Bezeichnung der Erfindung nicht angegeben ist, siehe Beschreibung.  
If no title is shown please refer to the description.  
Si aucun titre n'est indiqué se référer à la description.)

Data processing system and method for operating the same

In Anspruch genommene Priorität(en) / Priority(ies) claimed /Priorité(s)  
revendiquée(s)  
Staat/Tag/Aktenzeichen/State/Date/File no./Pays/Date/Numéro de dépôt:

Internationale Patentklassifikation/International Patent Classification/  
Classification internationale des brevets:

G06F12/00

Am Anmeldetag benannte Vertragsstaaten/Contracting states designated at date of  
filing/Etats contractants désignées lors du dépôt:

AT BE BG CH CY CZ DE DK EE ES FI FR GB GR IE IT LI LU MC NL PT SE SK TR

Data processing system and method for operating the same

EPO - DG 1

04.10.2002

99

The invention pertains to a data processing system having a hierarchical memory organization. The hierarchical organization of the memory serves to bridge the gap between fast processor cycle time and slow memory access time. Typically such a memory hierarchy comprises a relatively small but fast first level cache (highest ranked) coupled to the processor, and a slower, but relatively large second level cache coupled to said first level cache. The next lower ranked level may be a main memory but it may alternatively be a further, larger cache between the second level cache and the memory. At the lowest ranked level the memory hierarchy has for example a mass storage medium as a magnetic or an optical disc. Otherwise the main memory may be provided with data via a transmission system, such as a network or a modem connection. A more detailed description of some of the basic concepts discussed in this application is found in a number of references, including Hennessy, John L., et al., "Computer Architecture--A Quantitative Approach" (Morgan Kaufmann Publishers, Inc., San Mateo, Calif., 1990). Hennessy's text, particularly Chapter 8, provides an excellent discussion of cache memory issues addressed by the present invention.

Many cache controllers employ a "write-allocate", also referred to as "fetch-on-write" scheme. That means that on a write miss a full cache line is fetched from memory, inserted in the cache, and the addressed word is updated in the cache. This line remains in the cache for some time, anticipating further writes in the same area. This scheme is generally chosen as it hopefully reduces the amount of memory traffic, since most writes will hit in the cache and do not directly generate memory traffic. This scheme requires an administration of 2 bits per cache line (besides the data content and the address tag), to indicate that the address tag is (in)valid and that the data content is clean or dirty (has been written to). A 'dirty' cache line is written back to memory when the cache wants to re-use that location for a new word, or (exceptionally) when an explicit 'flush' operation is executed by the processor.

When operating on streaming (multi-media) data, tasks normally distinguish input and output streams. Output streams are written by the processor to a memory-based buffer. For such streams (such writes) the 'fetch-on-write' policy generates useless memory traffic (causing power dissipation and time delay) by first fetching the memory block into the cache. Clearly this read data is useless and will all be overwritten.

A known technique to avoid 'fetch-on-write' for all situations is to add more bits to the cache administration: a 'valid' bit for each data byte of every cache line. Per cache line, this set of 'valid' bits now replaces the single 'dirty' bit. Such a processing system, in accordance with the opening paragraph, is known from US 5,307,477. Upon a write miss, a new cache line is 'allocated' (it is given a location in the cache, the address tag and 'valid' bit are set), and the write operation is completed by inserting only the actually written bytes and setting the corresponding 'valid' bits. When this cache line needs to be flushed to memory, only part of its data might be 'valid', which is normally served by a memory system which allows a 'byte dirty mask' to be specified with memory write operations. Clearly this scheme totally avoids the 'fetch-on-write', at the cost of an extensive cache administration (valid bit per byte) and a more elaborate memory interface (having additional wires to transmit a 'dirty mask').

It is a purpose of the invention to reduce such useless 'fetch-on-writes', while maintaining the total amount of auxiliary data modest in comparison to the total size of the memory hierarchy. According to the invention the data processing system is characterized by the characterizing features of claim 1.

In the data processing system according to the invention useless fetch-on-writes are avoided, as the cache controller of the lower ranked cache recognises from the write mask whether the higher ranked cache replaces a subset of a cache line in the lower ranked cache, or replaces an entire cache line. If a subset of the cache line is to be replaced it is necessary to first fetch that cache line from M, if it was not already available in the lower ranked cache. However, if the higher ranked cache replaces an entire cache line in the lower ranked cache, the cache controller of the lower ranked cache recognises this from the writemask and avoids an unnecessary fetch on write. As the fetch on write is avoided in those cases the operational speed of the data processing system as a whole is improved. Although the higher ranked cache maintains a relatively extensive administration, the total administration overhead is modest, as the auxiliary information in the lower ranked cache concerns data elements at a coarser granularity than that in the lower ranked cache, and because the higher ranked cache is smaller than the lower ranked cache.

The invention is in particular advantageous to multiprocessor systems as claimed in claim 2. In this way the frequency that two processors need to simultaneously access the shared memory is reduced. This contributes to the operational speed of the multiprocessor system. In order to achieve the best results, preferably, each of the processors in the multiprocessor system has its own memory hierarchy as defined in claim 1.

The ratio between the line size of the lower ranked cache and the higher ranked cache may be an integer multiple greater than one, for example two. In that case, if the higher ranked cache writes a first line in the line of the lower ranked cache for which the writemask indicates that its content is fully replaced, and a second line for which the writemask indicates that its content is partially replaced, then it is necessary to fetch an entire line in said lower ranked cache first. The data which is fetched then also includes the data in the addresses of the next lower ranked level corresponding to the first line. In the preferred embodiment of claim 3 it suffices to fetch only the data corresponding to the second line in the lower ranked cache.

In the embodiment of claim 4 it is prevented that data traffic to the processor can interfere with data traffic from the processor. In addition to the higher ranked write cache the data processing system may have a separate read cache at the same rank.

The embodiment of claim 5 is a very cost-effective solution. Simulations have shown that even in a processing system according to the invention wherein the higher ranked cache has only one line results in a substantial reduction in the communication bandwidth of the background memory.

These and other aspects of the invention are described in more detail with reference to the drawing. Therein

Figure 1 schematically shows a first embodiment of a data processing system suitable for implementation of the invention,

Figure 2 shows a portion of Figure 1 in more detail,

Figure 3 schematically illustrates a method of operating a data processing system according to the invention,

Figure 4 schematically shows a second embodiment of a data processing system suitable for implementation of the invention.

A higher ranked cache C1 in the memory hierarchy has a cache controller CC1 operating according to a write allocate scheme, and a lower ranked cache C2 is coupled to the higher ranked cache C1 and has a cache controller CC2. In the embodiment shown the higher ranked cache is also the highest ranked (first level) cache C1, and the lower ranked cache is second level cache C2. However, for the purpose of the invention it is sufficient that the

higher ranked cache and the lower ranked cache are two mutually successive caches in the memory hierarchy, for example the second and the third level cache. Usually the highest level cache is direct mapped, or set associative with small sets. This contributes to response speed. Lower level caches usually are set associative with relatively large sets, so as to  
5 increase the hit-rate.

---

The size of the higher ranked cache C1 is smaller than the size of the lower ranked cache C2. Both caches administrate auxiliary information indicating whether data present therein is valid.

The lower ranked cache on its turn is coupled to a main memory M.

10 In order to illustrate the invention the lower C2 and the higher ranked cache C1 are shown in more detail in Figure 2.

In the embodiment shown the higher ranked cache C1 has 4 lines. Each line comprises a tag info unit, which indicates the memory address corresponding to the data in said cache line. The data includes here 4 data elements D1. In addition the cache lines store  
15 auxiliary information V1 in the form of validity bits, one validity bit for each data element.

Typically, C1 would have a 'valid' bit V1 per byte in the L1 cache line (the granularity of processor write operations). A preferred embodiment would match Figure 2 with the choice of having a 'valid' bit in C2 at the granularity of C1 cache lines. An alternative embodiment could have 'valid' bits V2 in C2 at the granularity of (4-byte or 8-  
20 byte) words.

The data processing system of the invention is characterized in that, the linesize of the lower ranked cache C2 is an integer multiple of the linesize of the higher ranked cache C1. In the embodiment shown in Figure 2 the linesizes of the lower ranked cache C2 is twice that of C1, i.e. A line in C2 comprises twice the number of data elements as  
25 compared to a line in C1. The auxiliary information V1 in the higher ranked cache C1 relates to data elements at a finer granularity than that in the lower ranked cache C2. More in particular, in the higher ranked cache C1, the cache lines comprise auxiliary info (valid info) V1 indicating the validity of each data element D1. In the lower ranked cache the valid info V1 relates to the granularity of four data elements (corresponding to the size of a higher level  
30 cache line) D2. The higher ranked cache C1 is arranged for transmitting a writemask WM to the lower ranked cache C2 in conjunction with a line of data DL for indicating which data in the lower ranked cache C2 is to be overwritten at the finer granularity. The writemask WM is constituted from the 4 valid bits V1 that belong to one C1 cache line.

BEST AVAILABLE COPY

The operation of a data processing system according to the invention is further described with reference to Figure 3.

In step 1 the lower ranked cache C2 receives a cache line from C1.

5 In step 2 the cache controller of C2 verifies whether the tag of the cache line from C1 corresponds with a tag in one of the cache lines of the lower ranked cache C2.

If the required tag is found the cache controller of C2 continues with step 3, wherein a masked write is performed in the corresponding location in C2. I.e. each bit in the mask indicates whether a corresponding data element in the cache line from C1 is valid. If this is the case, it should overwrite the corresponding data element in C2. The data written in  
10 C2 need not be copied immediately to the next lower ranked level in the memory hierarchy. Instead, a write-back strategy may be used, wherein the data may be copied when it has to be replaced by other data with a new tag, or in the exceptional case of a flush operation.

If the tag is not found the cache controller continues with step 4 wherein a line in C2 is identified which may be replaced. Several strategies, which are outside the scope of  
15 the invention can be applied to select a line to replace. Well known is for example the least recently used (LRU) method. Otherwise a random selection method may be applied for example.

In step 5 the line selected for replacement is written to the next lower ranked level M in the memory hierarchy.

20 In step 6 it is verified whether the data in the received cache line DL will overwrite all data in the corresponding cache line or part thereof in the lower ranked cache C2. If this is the case the cache controller CC2 continues with step 3, wherein the received cache line DL is written in the lower level cache C2.

If not all data will be overwritten a line is fetched in step 7 from the third level  
25 M of the memory hierarchy at an address corresponding to the tag T1 of the cache line from C1.

After step 7 the cache controller CC2 continues with step 3 wherein the received cache line DL is actually written in the selected cache line of the lower ranked cache C2.

30 Figure 4 shows a second embodiment of the invention, wherein the data processing system comprises one or more further processors P, P', P'', and wherein the memory hierarchy C1, C2, M of processor P comprises a memory having a rank which is lower than the rank of said lower ranked cache and which is shared with said other processors. In the embodiment shown each of the processors in the multiprocessor system has

its own memory hierarchy, hierarchy of processor P' comprising higher ranked cache C1' and lower ranked cache C2'. The hierarchy of processor P'' comprises higher ranked cache C1'' and lower ranked cache C2''.

BEST AVAILABLE COPY



## CLAIMS:

04.10.2002

(99)

1. A data processing system comprising  
- a processor and a memory hierarchy, wherein the highest ranked level in the hierarchy is a cache coupled to the processor, wherein  
- a higher ranked cache in the memory hierarchy has a cache controller  
5 operating according to a write allocate scheme,  
- a lower ranked cache is coupled to the higher ranked cache and has a cache controller,

wherein the size of the higher ranked cache is smaller than the size of the lower ranked cache,

10 wherein both caches administrate auxiliary information indicating whether data present therein is valid,  
characterized in that,

the linesize of the lower ranked cache is an integer multiple of the linesize of the higher ranked cache, wherein the auxiliary information in the higher ranked cache concerns data  
15 elements at a finer granularity than that in the lower ranked cache and wherein the higher ranked cache is arranged for transmitting a writemask to the lower ranked cache in conjunction with a line of data for indicating which data in the lower ranked cache is to be overwritten at the finer granularity, the cache controller of the lower ranked cache being  
20 arranged for fetching a cache line from the next lower ranked level in the memory hierarchy if that line is not cached yet and the writemask indicates that the data in the line provided by the higher ranked cache is only partially valid, and wherein fetching a line from said next lower ranked level is suppressed if the writemask indicates that the line provided by the higher ranked cache is valid in accordance with the courser granularity of the auxiliary information in the lower ranked cache, in which case, the controller of the lower ranked  
25 cache allocates the cache line in the lower ranked cache without fetching it.

2. Data processing system according to claim 1, comprising one or more further processors, and wherein the memory hierarchy comprises a memory having a rank which is

lower than the rank of said lower ranked cache and which is shared with said other processors.

3. Data processing system according to claim 1 or 2, wherein the cache lines of  
5 the lower ranked cache and the higher ranked cache have the same number of data elements.

4. Data processing system according to one of the previous claims, wherein the higher ranked cache is a write only cache.

10 5. Data processing system according to one of the previous claims, wherein the higher ranked cache has exactly one cache line.

6. Method for operating a data processing system comprising a processor and a memory hierarchy, wherein the highest ranked level in the hierarchy is a cache coupled to the  
15 processor, wherein

- a higher ranked cache in the memory hierarchy has a cache controller operating according to a write allocate scheme,

- a lower ranked cache is coupled to the higher ranked cache and has a cache controller,

20 wherein the size of the higher ranked cache is smaller than the size of the lower ranked cache,

wherein both caches administrate auxiliary information indicating whether data present therein is valid,  
characterized in that,

25 the linesize of the lower ranked cache is an integer multiple of the linesize of the higher ranked cache, wherein the auxiliary information in the higher ranked cache concerns data elements at a finer granularity than that in the lower ranked cache,  
according to which method

- the higher ranked cache transmits a writemask to the lower ranked cache in  
30 conjunction with a line of data for indicating which data in the lower ranked cache is to be overwritten at the finer granularity,

- the cache controller of the lower ranked cache fetches a cache line from the next lower ranked level in the memory hierarchy if that line is not cached yet and the

writemask indicates that the data in the line provided by the higher ranked cache is only partially valid, and

- wherein fetching a line from said next lower ranked level is suppressed if the writemask indicates that the line provided by the higher ranked cache is valid in accordance with the courser granularity of the auxiliary information in the lower ranked cache, in which case, the cache controller of the lower ranked cache allocates the cache line in the lower ranked cache without fetching it.

5

## ABSTRACT:

04.10.2002



A data processing system according to the invention comprises a processor (P) and a memory hierarchy. The highest ranked level therein is a cache coupled to the processor. The memory hierarchy comprises a higher ranked cache (C1) having a cache controller (CC1) operating according to a write allocate scheme, and a lower ranked cache (C2) is coupled to the higher ranked cache (C1) having a cache controller (CC2). The size of the higher ranked cache is smaller than the size of the lower ranked cache. Both caches (C1, C2) administrate auxiliary information (V1, V2) indicating whether data (D1, D2) present therein is valid. The linesize of the lower ranked cache (C2) is an integer multiple of the linesize of the higher ranked cache (C1). The auxiliary information (V1) in the higher ranked cache (C1) concerns data elements (D1) at a finer granularity than that in the lower ranked cache (C2). The higher ranked cache (C1) is arranged for transmitting a writemask (WM) to the lower ranked cache (C2) in conjunction with a line of data (DL) for indicating which data in the lower ranked cache (C2) is to be overwritten at the finer granularity. Fetching a line from the next lower ranked level (M) is suppressed if the writemask (WM) indicates that the line (DL) provided by the higher ranked cache (C1) is entirely valid in which case, the controller (CC2) of the lower ranked cache allocates the cache line in the lower ranked cache (C2) without fetching it.

Fig. 1

04.10.2002

99

1/3

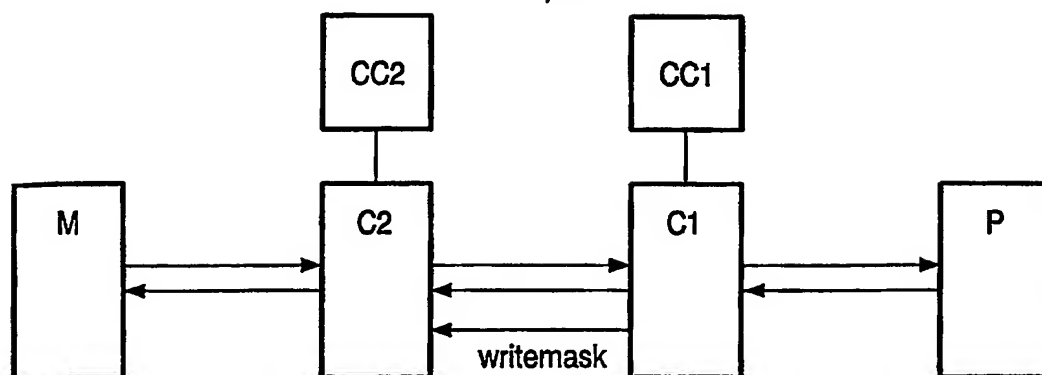


FIG. 1

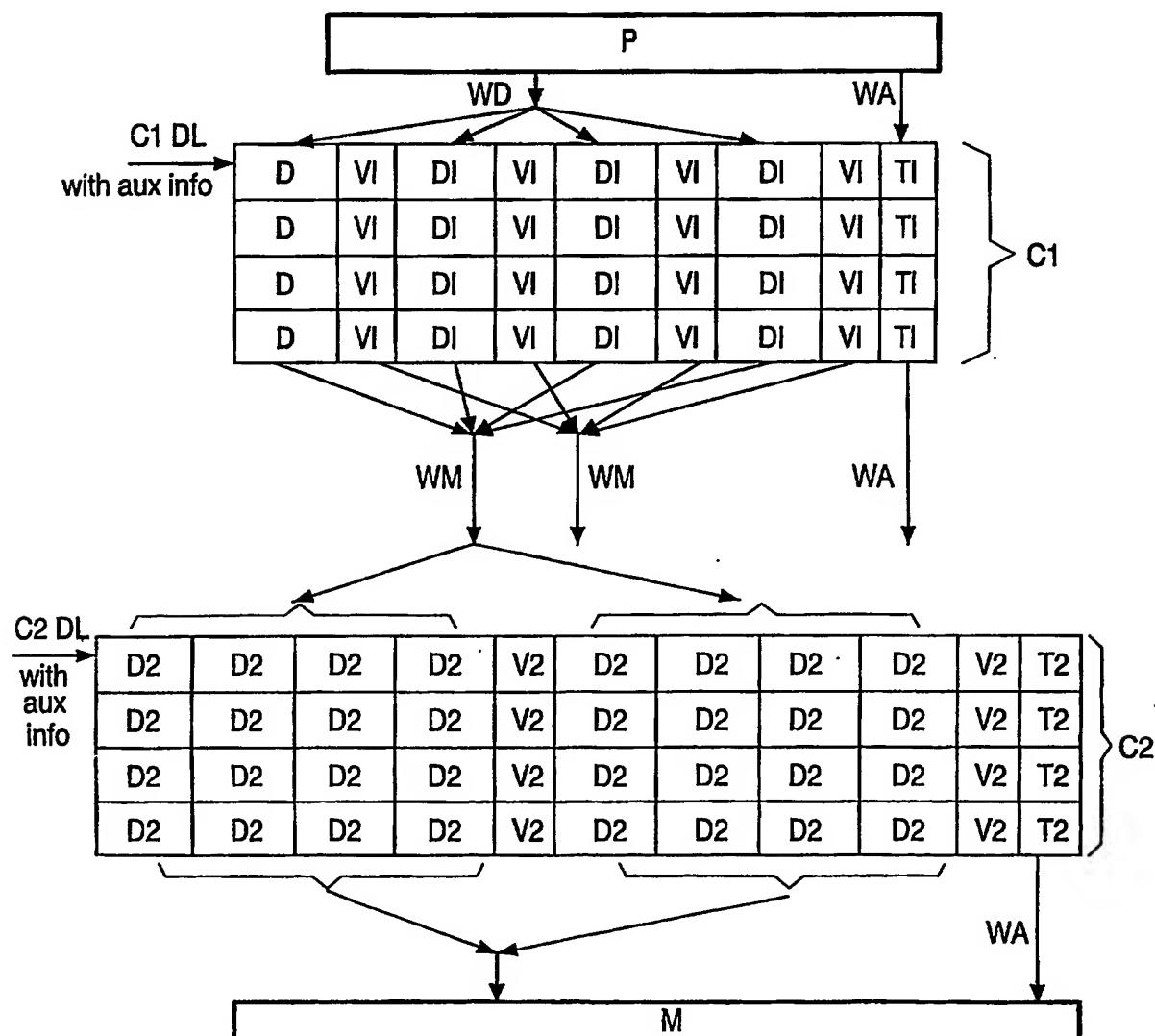


FIG. 2

2/3

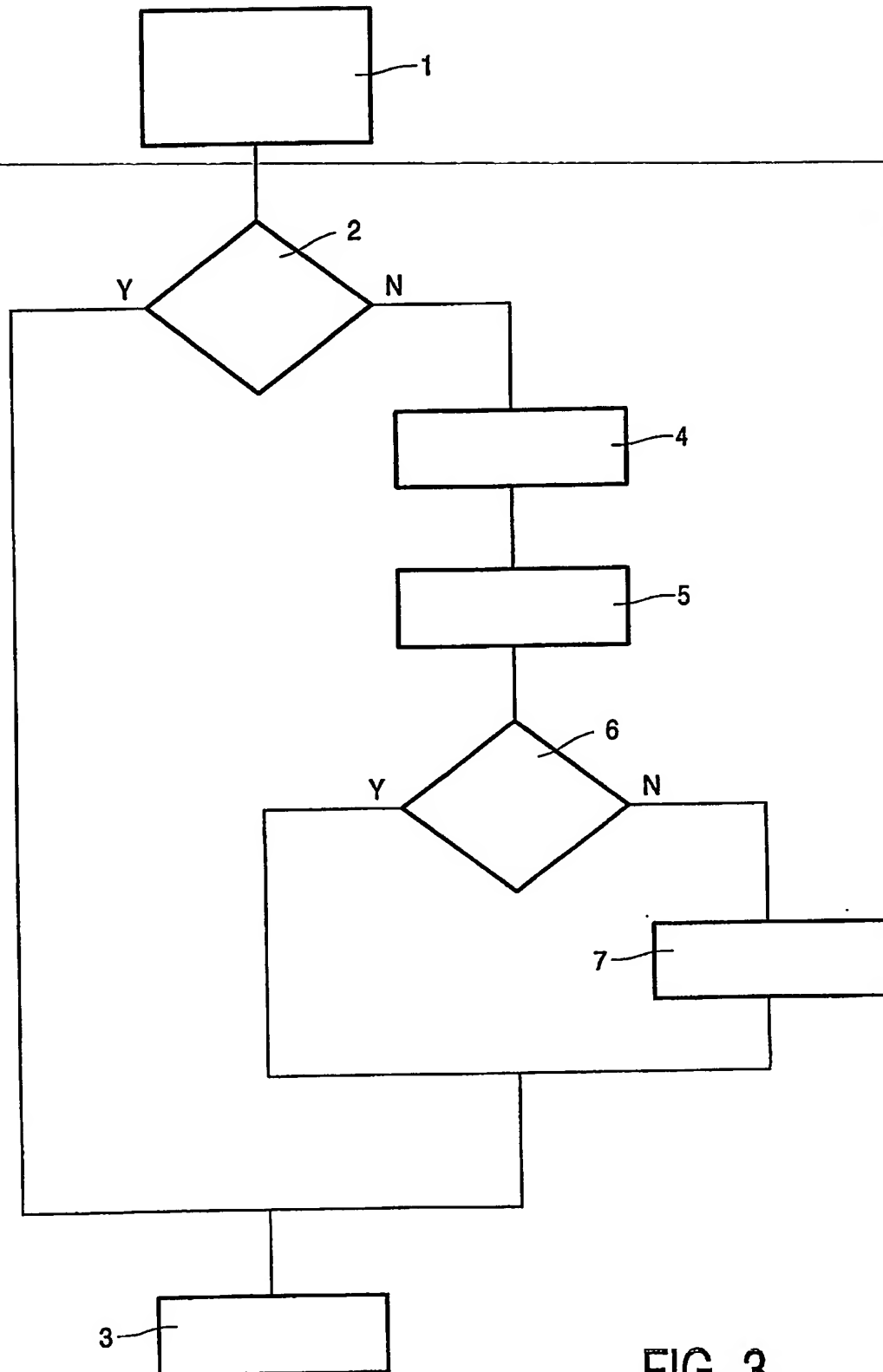


FIG. 3

BEST AVAILABLE COPY

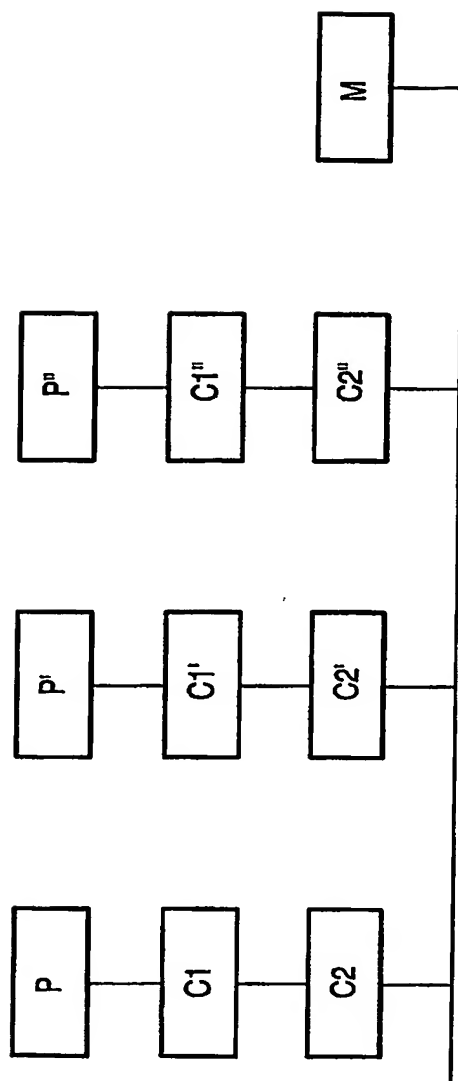


FIG. 4